

# App Lab を用いた初心者向けプログラミング授業の実践 文系大学生を対象としたプログラミング学習の導入について

榊原 直樹<sup>1</sup>

## Practice for Beginners Programming Lesson Using App Lab

### Introduction of Programming Learning for Undergraduate Students

*Naoki SAKAKIBARA*

#### Abstract

App Lab is an online programming education environment. It was designed classes of programming for beginners using the App Lab. Through 15 lessons of the class, it was to understand the basic structure of the programming of the sequential-repetition-branch. Students were allowed to complete the game as a final project. The effectiveness of App the Lab has been confirmed from these results.

キーワード：情報教育、プログラミング、code.org、App Lab、授業デザイン

Keywords：Information-related education , Programing , code.org , App Lab , class design

#### 1. はじめに

##### 1.1 プログラミング教育の重要性

文部科学省において小学校でのプログラミング教育の必修化が検討されるなど、今後ますますプログラミングを学ぶ機会が増えることが予想される。これまでエンジニアとなる人材を対象に作られてきたプログラミングに関する教材ではなく、一般の初心者に対応したカリキュラムが求められている。

##### 1.2 清泉女学院大学での IT 教育

清泉女学院大学人間学部心理コミュニケーション学科では入学者に対して、基本的な IT リテラシー教育を実施した後、ウェブデザインなどの専門科目を履修することになっている。ウェブデザインに関する科目はいくつかに分かれており、HTML5/CSS3 に関する科目を履修した後に、画像処理などの画面デザインを学ぶ科目や JavaScript を使ったウェブプログラミングの科目を履修し、最終的に一人でウェブサイトを完成させる能力を身につけることを目的としている。

HTML5/CSS3 や画像処理は複数の科目が連携して進んでおり、十分な習熟ができるように構成されているが、プログラミングについては 1 科目のみで時間がやや不足している。また他の科目との連携がなく、独立したカリキュラムで進んでいるためより効率的な学習計画が求められていた。

##### 1.3 プログラミング学習環境の選考

効率的にプログラミングを学習するために、まずは学習環境を見直すことにした。初心者向けのプログラム学習環境はさまざまなものがあるが、代表的なものとして MIT が開発した Scratch がある。ブロックを組み合わせるプログラミングを構築するビジュアル言語を採用しており、小学校の授業でも採

---

<sup>1</sup> 清泉女学院大学

用されるなど広く利用されていることから、教材やノウハウも豊富で講義を作成しやすいメリットもある。しかし Scratch は独自のスクリプト言語を採用しており、ウェブプログラミングで利用される言語とは異なる体系のものである。プログラミングの基礎概念を学ぶために Scratch を使用し、その後ウェブプログラミングに関するスクリプトを学ぶことなども検討したが、異なるスクリプト言語を短期間で切り替えて学習するのは効率が落ちると考えた。

次にウェブプログラミングに関する部分について学ぶときに使用する言語を選定する。フロントエンドの開発に用いる場合、現状では JavaScript を用いるのが一般的である。

サーバーサイドの場合はもう少し選択肢が広く Java などのコンパイル言語や、PHP/Ruby/Python などのスクリプト言語などが選択できる。様々な選択肢があるが、初心者がつまづきやすいポイントとして開発環境の構築が難しい事が挙げられる。そのためコンパイラやサーバーの設定が必要なサーバーサイドの開発環境は除外することにした。

結果として JavaScript を用いた講義を計画することになった。JavaScript は前述のテキストエディタで記述したものをブラウザで実行できるので、サーバーサイドのプログラミング環境に比べて環境構築が簡単であることが大きなメリットである。

一方で JavaScript はブラウザで動作するスクリプト言語なので、同時に HTML/CSS の知識が必要となる。清泉女学院大学ではこれらに関する講義を実施しているが、授業選択の都合で未履修の学生が参加する可能性があることも考慮して授業計画を立てる必要がある。

こうした状況を踏まえて、各種のプログラミング学習環境を比較した結果、米 NPO 団体の Code.org が提供する「Hour of Code」と「App Lab」を利用して授業計画を立てることにした。

### 1.3.1 Code.org

Code.org とは児童・生徒を主な対象としてプログラミング教育普及を目的にしているアメリカの NPO 団体である。2013 年に設立され、ウェブサイトを通じて初心者向けのオンライン教材を提供している。



図1 Code.org のウェブサイト

Code.org は様々な教材を提供しているが、もっとも有名なものが Hour of Code である。文字どおり 1 時間でコードの書き方を覚えることができるというコンセプトで作られた教材である。画面に配置されたキャラクターを操作するブロックを組み立てて、課題をクリアするゲーム仕立ての構成になっており、ブロックの組み立てを通じてプログラミングの概念を学ぶことができる。

ゲームのテーマには子どもたちに人気の映画「アナと雪の女王」や「スターウォーズ」などが採用

されており、それぞれの作品の登場人物を操作することで楽しくプログラミングを学習できるように配慮されている。



図2 Hour of code の画面

App Lab も code.org が提供する教材である。Hour of code が K-12 を対象に設計されているのに対して、App Lab は高校から大学 1.2 年程度を想定する設計されている。

Hour of code があらかじめ決められたシチュエーションの問題を解決するためにプログラムを作成するのにに対して、App Lab では自分で課題を設定してコードを書きプログラムを完成させる。このため自由にプログラミングができるようになるためには、ある程度の習熟が求められる。

プログラム言語は JavaScript を採用しており、これを Google 社が開発した Blockly というビジュアルエディターを使って記述することができる。Hour of code と似た画面操作なので、経験者はすぐに操作を覚えられるメリットがある。プログラムの入力画面は 3 ペインで左からアプリの動作画面、プログラムに用いる Blockly のツール画面、コードを記述するエディター画面に分かれる。

アプリの動作画面はスマートフォンを模しており、実際に完成したアプリをスマートフォンのブラウザ上で動作させることも可能である。



図3：App Lab の編集画面 コード編集画面（左） デザイン編集画面（右）

Blockly のツールは「UI Control」、「Data」、「Control」、「Variables」、「Canvas」、「Turtle」、「Math」、「Function」の 8 つに分類されており、タブを切替えてブロックを呼び出すことが可能である。

エディター画面はツール画面から選んだブロックをドロップすることでプログラムを記述できる。エディター画面はブロックとテキストの表示を切替えることが可能であり、テキストモードに設定すると、プログラムを JavaScript で表示することが可能である。

画面を「Code」モードから「Design」モードに切替えると、アプリの動作画面に配置するボタンや

画像をWYSIWYGで編集することができる。Designモードで配置したボタンや画像はIDで管理され、タップした際の動作を指定することが可能である。

編集モードにはこのほかに「Data」がある。App Labには簡易的なデータベース機能が用意されており、サーバーに保存されたデータテーブルを編集する事が可能である。

## 2. 授業計画

### 2.1 目標設定

授業を通じて学生に理解してもらいたい概念として、次の3つを挙げた。1. 順次処理、2. 繰り返し、3. 条件分岐。これらを4つの段階を通じて繰り返し学習することにした。

### 2.2 コース設計

これまでプログラミングを体験したことのない学生に対して、効率よくプログラミングを身につけるために、導入・基礎・展開・応用の4段階に分けて授業設計を行った。

#### 2.2.1 導入

初めてプログラミングを学ぶ学生が気軽に学習を始められるように、Hour of Codeから開始することに。Hour of codeは多くのゲームが用意されているが、その中から先に挙げた3つの概念を取り入れたゲームを選択し、ゲーム実施後にそれぞれの概念について解説を行った。

#### 2.2.2 基礎

次にApp Labに移って開発環境の開設を行い、そのうえでプログラミングの基礎知識について解説した。ここではHour of codeで行ったゲームを例に、それをプログラムで実現するにはどうすればよいかについて解説した。

#### 2.2.3 展開

解説した3つの概念（順次処理、繰り返し、条件分岐）を使ってプログラミングを行った。課題として、条件分岐で結果が変わるアドベンチャーゲームや、入力したデータを変数にして行う占いゲームなどを作成させ、概念の定着をはかった。

#### 2.2.4 応用

最後に展開で作成した課題を基に、オリジナルのゲームプログラムを作成おこなうことにした。ゲームはスマートフォンで操作できるようにデザインすることにした。

## 3. 授業評価

### 3.1 授業結果

当初の予定通りのスケジュールですべての講義を終了することができた。受講生は14名で全員が課題を提出することができた。途中で脱落することなく最後まで続けることができたことは、導入の段階でプログラミングに対する苦手意識を持たず、ゲーム作成などの課題を各自の創意で楽しく進めることができたことが大きい。

講義中に課した問題で理解度が高かったのが、条件分岐や繰り返し処理の概念だった。これはBlocklyを実行する際に、処理の経過を視覚的に表示する機能があり、プログラムの流れを直感的に把握しやすいことがあげられる。

これに対し変数の概念はデータを一時的に保存するイメージを、他の概念にくらべ視覚的に捉えに

くいため、質問等が多かった。これについてはアンプラグドによる説明で補足を行った。アンプラグドとはコンピューターを使わずにコンピューターの仕組みやプログラミングの概念を学ぶ手法である。

### 3.2 課題の結果

講義の仕上げとして各自でプログラミングしたゲームを完成させる課題を出した。次に示すのは受講者が作成したプログラムの動作画面をキャプチャーしたものである。

図4（左）はゴミの分別を学習するクイズ形式のアプリである。画面上部に表示されたゴミの分別方法を画面下部に表示された2つの分別方法から選択して答えるものである。実際の自治体のルールに基づいてクイズを作成すれば環境教育に応用できる可能性がある。

図4（中）は選択肢に合わせてストーリーが変わるロールプレイングゲームである。場面に応じて画像や音楽が変化する。ストーリーは短いが、ゲームとして成立している。

図4（右）はモグラたたきゲームである。画面上のモグラを一定時間の間に何買いたたけるかを競うもので、叩いた回数に応じてエンディングが変わる工夫がしてある。

いずれのプログラムも講義の中で説明した機能のみで完成させており、概念が十分に定着していることが分かった。



図4 課題で作成したゲーム

ゴミの分別アプリ（左）

選択肢を選んで進むロールプレイングゲーム（中）

モグラたたきゲーム（右）

## 4. 今後の展開

### 4.1 改善点

App Labを用いたプログラミングの講義は初心者にとって、プログラミングの概念を理解するために有効であることが分かった。一方で講義を通じていくつかの問題点も発見された。

App Labで使用しているBlocklyはJavaScriptだが、Webページで使われる一般的なJavaScriptとは関数などで仕様が異なる部分があった。ウェブプログラミングとしてJavaScriptを使用するにはこ

これらの差異について補足する必要がある。

また日本語の扱いについてはプログラム中で使用するには多言語対応しており、問題なく使用することができた。しかしコメント文に日本語を入れると文字化けを起こすなどの問題が見られた。また翻訳の一部に不備が見られた。これらの問題は `code.org` にフィードバックして今後改善していくことが必要である。

講義で説明した3つの概念（変数、繰り返し、条件分岐）で基本的なプログラミングを行ったが、これらを用いて、より複雑な処理ができるようにアルゴリズムについて解説をすることが次の段階として必要だと考えられる。

## 4.2 応用

App Lab の中で使用されている Blockly はオープンソースソフトとして公開されている。これを組み込んだ開発環境は他にもいくつかあり、代表的なものに MIT が運用しているオンラインのアプリ開発環境の App Inventor がある。App Inventor は Blockly で記述したプログラムを Android OS で動作するネイティブなアプリを作成することができる。App Lab でも作成したアプリをスマートフォンで動作させることができるが、App Inventor ではネイティブアプリを作成できるため、スマートフォンに搭載されているカメラや加速度センサ、GPS 機能などをプログラムの中で利用できるため、プログラムの応用範囲が広がる。作成したプログラムの実行に Android 端末もしくは PC 上にエミュレータのインストールが必要であり管理者権限などの関係で開発環境の構築に課題があること。また本学の学生の多くが iOS の端末を使用していることなどが検討課題に挙げられる。

Blockly にはエディター部分を独立させた「Blockly for Web」がある。これは JavaScript の他、Python など5つの言語に対応しており、JavaScript をきっかけにその他の言語を同時に学ぶことが可能であることもメリットである。

## 5. 結び

初心者向けに開発されたプログラミング教材である App Lab を用いた講義は完成した課題の水準から効果的であると判断した。段階的に学習を進めることで初心者でも短時間でプログラミングの概念を理解することが可能であることが分かった。本講義で取り上げた3つの概念（順次処理・繰り返し・条件分岐）を利用すれば、スクリプト言語やマクロ機能などを使って仕事の生産性を上げることが可能であり、文系の学生にとっては十分な成果と考える。システムエンジニアを目指す学生には、この講義だけでは不十分で、より実践的な学習が必要である。今後の課題は、より高度な概念や処理を理解し、実際のプログラミングへつなげていくことである。

## 6. 謝辞

ウェブプログラミング演習を受講してくれた学生達ならびに査読を引き受けてくださった長野県短期大学多文化コミュニケーション学科 萱津理佳先生に感謝します。

（受付日：2017年3月15日）